**BỘ THÔNG TIN TRUYỀN THÔNG**



**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

----------

**Kỹ thuật giấu tin trong văn bản**

**Sử dụng CFG để giấu tin vào văn bản và truyền tin**

**Sinh viên:** Lê Văn Tuấn

**Mã sinh viên:** B21DCAT208

**Bộ môn:** Kỹ thuật giấu tin

**Giảng viên hướng dẫn:** PGS.TS. Đỗ Xuân Chợ

**Hà Nội – 2025**

**MỤC LỤC**

[**MỤC LỤC** 2](#_Toc192174102)

[**I.** **Giới thiệu chung về bài thực hành** 3](#_Toc192174103)

[**II.** **Nội dung và hưỡng dẫn thực hành** 3](#_Toc192174104)

[**III.** **Phân tích yêu cầu bài thực hành** 3](#_Toc192174105)

[**IV.** **Thiết kế bài thực hành** 3](#_Toc192174106)

[**V.** **Cài đặt và cấu hình máy ảo** 3](#_Toc192174107)

[**VI.** **Tích hợp và triển khai** 3](#_Toc192174108)

[**VII.** **Thử nghiệm và đánh giá** 3](#_Toc192174109)

1. **Giới thiệu chung về bài thực hành**

Ở bài thực hành này, sinh viên sẽ được làm quen với phương pháp sử dụng văn phạm phi ngữ cảnh. Sử dụng một văn phạm phi ngữ cảnh (CFG-Context Free Grammar) để sinh ra các câu tạo thành văn bản nhân tạo chứa thông điệp bí mật và văn bản này có thể bắt chước văn bản thực tế. Văn phạm phi ngữ cảnh là một tập hợp hữ hạn các biến (còn được gọi là các ký hiệu chưa kết thúc), mỗi biến biểu diễn một ngôn ngữ. Ngôn ngữ được biểu diễn bởi các biến được mô tả một cách đệ quy theo thuật ngữ của một khái niệm khác gọi là ký hiệu kết thúc. Quy tắc quan hệ giữa các biến gọi là luật sinh. Mỗi luật sinh. Mỗi luật sinh có dạng một biết, ở vế trái sinh ra một chuối có thể gồm biến lẫn các ký hiệu là văn phạm G (V, T, P, S), trong đó:

* V là tập hợp hữu hạn các biến
* T là tập hợp hữu hạn các ký tự kết thúc V ∩ T = ∅
* P là tập hợp hữu hạn các luật sinh mà mỗi luật sinh có dạng A → α (với A là biến và α là chuỗi các ký hiệu ∈ (V∪T)\*)
* S là một biến đặc biệt gọi là ký hiệu bắt đầu văn phạm.

Các quy tắc sau chỉ ra cách để tạo ra một chuỗi (gồm các kí hiệu kết thúc) từ một CFG cho trước:

* Sử dụng 1 kí hiệu bắt đầu để khởi đầu. Chọn một luật có kí hiệu khởi đầu ở bên trái và chọn một kí hiệu ở phần bên phải của luật để thay thế kí hiệu khởi đầu này. Kí hiệu được sử dụng sẽ là một thành tố của văn bản sẽ được tạo sao này.
* Chọn 1 kí hiệu chưa kết thúc trong văn bản, tìm một luật có kí hiệu chưa kết thúc này ở bên trái và thay thế kí hiệu này bằng một kí hiệu ở phần bên phải của luật.
* Lặp lại bước 2 cho đến khi văn bản chỉ còn lại toàn các kí hiệu kết thúc.

Sau khi giấu tin xong sẽ gửi văn bản đến một máy client sau đó client giải mã thông điệp.

1. **Nội dung và hưỡng dẫn thực hành**
   1. **Mục đích**

Giúp sinh viên hiểu được quá trình giấu tin trong văn bản bằng phương pháp CFG và cách giải mã để trích xuất tin được giấu trong văn bản.

* 1. **Yêu cầu người thực hành**

Sinh viên có kiến thức cơ bản về mật mã cơ sở, giao thức ssh và có khả năng sử dụng lệnh linux cơ bản

* 1. **Nội dung thực hành**

Sinh viên khởi động bài lab

Chạy lệnh

*labtainer -r cfg-lab*

*(Chú ý: Sinh viên sử dụng <Tên\_tài\_khoản> của mình để nhập thông tin người thực hiện bài lab khi có yêu cầu, để sử dụng khi chấm điểm)*

Sau khi khởi dộng lab xong, 2 container hiện lên và sinh viên thực hiện làm theo các nhiệm vụ:

**Task 1:**

Ở đây, Sinh viên có thể thấy 2 máy là client1 và client2, trong máy client1 có *sender.py* và *cfg.txt* lần lượt là chương trình python của người gửi để giấu một thông điệp vào trong văn bản nhân tạo và cfg.txt chứa các luật sinh cho *sender.py.*

Máy client2 cũng sẽ có 1 file cfg.txt chứa luật sinh và receiver.py để trích xuất thông điệp được giấu trong văn bản được gửi.

Mục tiêu của sinh viên là gửi được thông điệp **“yes”** được giấu trong văn bản và máy client2 giải mã thành công.

Dùng *ifconfig* ở máy client2 để kiểm tra địa chỉ IP

*ifconfig*

Thực hiện giấu thông điệp “yes” vào văn bản bằng lệnh

*python3 sender.py -m <thông\_điệp\_cần\_giấu>*

Nếu thực hiện đúng, văn bản giấu tin sẽ được ghi vào file message.txt

**Task 2:**

Sau khi thu được văn bản giấu tin, tiến hành gửi nó cho máy client2 bằng ssh, sử dụng câu lệnh:

*scp message.txt client2@<IP\_client2>:/home/client2*

Nếu thành công, bên máy client2 sẽ xuất hiện file message.txt.

**Task 3:**

Tiến hành đọc file message.txt từ máy client2. Đây là một câu đơn thuần, bình thường sẽ không thể nhận biết được thông điệp thực sự được giấu trong đó là gì. *cat message.txt*

**Task 4:**

Bây giờ thực hiện giải mã văn bản giấu tin ở máy client2 bằng file receiver.py

*python3 receiver.py*

Nếu thành công, thông điệp sẽ được giải mã và in lên màn hình

Kết thúc lab:

* Trên terminal khởi động lab, sinh viên sử dụng lệnh:

*stoplab*

* Khi bài lab kết thúc, một tệp lưu kết quả đuọc tạo và lưu vào một vị trí được hiển thị bên dưới stoplab. Sinh viên cần nộp file .lab để chấm điểm.
* Để kiểm tra kết quả trong khi làm bài thực hành sử dụng lệnh:

*checkwork <tên bài lab>*

* Sinh viên cần nộp file .lab để chấm điểm.
* Kiểm tra kết quả trong quá trình làm bài:

*checkwork <tên bài lab>*

* Khởi động lại bài lab: Trong quá trình làm bài sinh viên cần thực hiện lại bài lab, dùng câu lệnh:

*labtainer -r cfg-lab*

1. **Phân tích yêu cầu bài thực hành**
2. **Thiết kế bài thực hành**
3. **Cài đặt và cấu hình máy ảo**
4. **Tích hợp và triển khai**
5. **Thử nghiệm và đánh giá**